Post print by authors.

An Improved Phenotype-Genotype Mapping for Solving Selective Assembly Problem Using Evolutionary Optimization Algorithms

Abolfazl Rezaei Aderiani*[1], Kristina Wärmefjord[1], Rikard Söderberg[1]
[1] Department of Industrial and Materials Science, Chalmers University of Technology, Gothenburg, Sweden.

Selective assembly is an assembly technique for producing high-quality assemblies from relatively lower quality mating parts. Developing the application of this technique to sheet metal assemblies in the automotive industry can improve the geometrical quality and reduce production costs significantly. Nevertheless, the required calculation time is the main obstacle against this development. To apply a selective assembly technique, an optimization problem of finding the optimal combination of mating parts should be solved. This problem is an MINLP optimization problem for selective assembly of sheet metals. This paper demonstrates that the phenotype-genotype mapping commonly used in most conventional selective assembly studies enlarges the search domain of the optimization. Thereafter, a new approach that makes the mapping one-to-one is proposed and applied to three selective assembly sample cases from the literature. Moreover, it is indicated that meta-heuristic methods are superior to MILP and MINLP methods in solving this problem, particularly for assemblies of more than two components and relatively large batch sizes. The results evidence that using the new method improves the convergence rate of meta-heuristics in solving the problem by reducing the number of cost function evaluations to 45% for sheet metal assemblies. This means reducing up-till 26 hours of the optimization time for the presented sample cases.

1 Introduction

Selective assembly can improve the geometrical quality of assembled products without tightening production tolerances of mating parts. Although implementing this technique started in the 1950th [1], it is gaining more attention in new production systems such as cyber factories and digital twins used in real-time to optimize production quality. For instance, implementing a new digital twin in production phase is proposed by Söderberg et al. [2] that utilizes a selective assembly technique. Another example is the utilization of a selective assembly technique in cyber factories by Colledani et al. [3]. Utilizing a digital twin in production of assemblies, can considerably reduce the cost of production and improve the geometrical quality of the assemblies [4, 5]. Figure 1 represents a schematic illustration of using selective assembly in a digital twin based assembly line for sheet metal assemblies. The produced parts for the assembly process are scanned firstly. Then, a model of each part including its geometrical deviations is generated as its digital twin. These models are used to simulate the assembly process to predict the geometrical deviations of the assemblies after releasing the fixture clamps and spring back. Therefore, an optimization algorithm can be utilized along the simulation tools to find the optimal combination of parts so that the geometrical quality of the assemblies is maximum. This combination of parts is used to fabricate the physical assemblies. Afterward, by scanning the physical assemblies, the obtained geometrical quality from the physical assemblies can be compared with the predicted geometrical quality by the digital twins and using reinforcement learning techniques errors of the simulations can be improved [6].

Sheet metal assemblies are the dominant type of assemblies in the automotive industry. There are different methods of reducing the geometrical variations in these assemblies, including adding slip-planes in joints to reduce the effects of geometrical variations in components [7, 8]. However, for spot-welded sheet metals, selective assembly technique is a promising alternative.

To apply selective assembly technique to spot-welded sheet metal assemblies, non-rigid variation simulations are required [9]. These simulations utilize finite element methods to predict the geometrical deviations of assemblies given information about part deviations, weld properties and other effective factors during the assembly process. The simulations contain non-linear finite element analysis and they are quite time-consuming for common types assemblies in the automotive industry. There have been some improvements to reduce the computation costs of these simulations [10, 11]. Nevertheless, utilizing them along an optimization algorithm to find the optimal combination of parts for performing selective assembly requires thousands of simulations to find the optimal combination of parts for each batch. Utilizing selective assembly technique in a digital twin based assembly line requires real-time control of assemblies. Hence, time or

*Corresponding Author: Email: aderiani@chalmers.se

computation cost is the main obstacle against using selective assembly techniques in a digital twin setup in the automotive industry [9].

Evolutionary optimization algorithms such as Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) are common tools in solving selective assembly problems. There is one mapping method to transfer the phenotype (selective assembly optimization parameters) to genotype (GA individual solutions) that has been used commonly in all studies that have utilized evolutionary optimization algorithms to solve selective assembly problems. This study demonstrates that meta-heuristic methods are superior in solving these problems and the utilized mapping in these algorithms can be improved to increase considerably the performance of optimization and accordingly the optimization time. Therefore, an introduction to selective assembly technique and studies that have focused on it is given in Section 1.1. Then, the existing gap in these studies and the scope of this paper in filing that gap is presented in Section 1.2.

1.1 Selective assembly

The primary method of executing selective assembly that is still common in production of engines and bearings is the following procedure. In this paper, the word "components" refers to the elements of an assembly and "parts" refers to produced parts for mass production of that element. Firstly, the dimensions of all produced parts will be measured because the dimensions vary between the allowed tolerance limits. Secondly, these parts will be divided into several groups (e.g. six groups) based on their measured dimensions. Figure 2 demonstrates this grouping when the measured dimensions are normally distributed.

The third step is to assemble the first group of Component A with the last group of Component B, the second group of A with next to the last group of B and so on. But, as shown in Figure 2, the number of parts in the matching groups may not be equal. For instance, the number of parts in the first group of Component A is not equal to the number of parts in the final group of Component B. Therefore, some parts would be useless. The early studies in selective assembly are mostly focused on overcoming this problem [1, 12–16].

A selective assembly problem can be categorized as an assignment problem. Coullard et al. [17] provided a linear time algorithm to solve maximum cardinality problems for assembly of bearings. Iwata et al. [18] developed algorithms for selective assembly of two different types of components. Considering selective assembly as an assignment problem, the workers and tasks can be defined as parts of the first and the second components, respectively. The cost of the workers should be defined as the deviation of the KPC in each assembly. Hence, the problem can be solved using Hungarian or Auction algorithms [19]. For assemblies of more than two components, a multidimensional Assignment problem should be solved. Tan and Wu [20] have presented a general formulation for direct selective assembly and fixed bin selective assembly and solved them using branch and bound methods.

By emerging evolutionary optimization algorithms, some studies utilized these algorithms to find the optimal combination of groups in selective assembly problems. When components are more than two or when sheet metal assemblies are involved, an optimization algorithm is required to find the optimal combination of groups or parts. For example, an assembly of three components with n parts of each component will produce n assemblies. The optimization problem is then to find the optimal combination of A_i, B_j and C_k, $i, j, k \in \{1, 2, \ldots, n\}$ so that the total variation of assemblies becomes minimal.

Kannan et al. [21] proposed a coding system from phenotype to genotype to solve the selective assembly problem using GA. In this method, a combination of groups (phenotype) is mapped to a chromosome (genotype, an individual solution in GA). The chromosome is thereafter divided into substrings, with each substring representing one component in the assembly. Since all components have the same number of groups, the length of substrings is equal to the number of groups. Therefore, a substring includes a sequence of integers in which each number represents a group number of that component. Consider, for instance, an assembly that includes three components and produced parts of those components are divided into six groups. As a result, a random combination of groups emerges as follows: $(A_6B_1C_5)$, $(A_2B_3C_6)$, $(A_1B_4C_1)$, $(A_3B_5C_2)$, $(A_4B_2C_4)$, and $(A_5B_6C_3)$. This phenotype will be mapped to the following genotype: 621345 134526 561243. The first six integers of this genotype represent the group numbers of the first component, the second six integers show the group numbers of the second component, and the final six integers represent the group numbers of the third component. The same type of coding is used for utilizing PSO to solve the selective assembly problem [22, 23].

Different methods for solving the selective assembly problem using evolutionary algorithms, including GA and PSO, have been developed. Kannan et al. [24] developed a three-stage optimization to minimize the surplus parts in selective assembly. Aderiani et al. [25] presented a multistage optimization process for components with any type of distribution in their dimension so that the surplus parts are zero and the variation is also minimal. All these studies utilized the same phenotype-genotype mapping that was developed by Kannan et al. [21].

1.2 Scope of paper

The phenotype-genotype mapping that has been developed by Kannan et al. [21] is used in other studies to the best of our knowledge [9, 22–31]. This paper demonstrates that this method of mapping is not one-to-one and consequently it increases the convergence time of the optimization. The time for finding the best combination of groups or parts is a critical parameter for utilizing selective assembly technique for sheet metal assemblies in smart assembly lines and cyber factories [9]. This is because in selective assembly of sheet metals each function evaluation requires running variation simulations of assemblies, and there is not usually a long time available between scanning the incoming parts un-

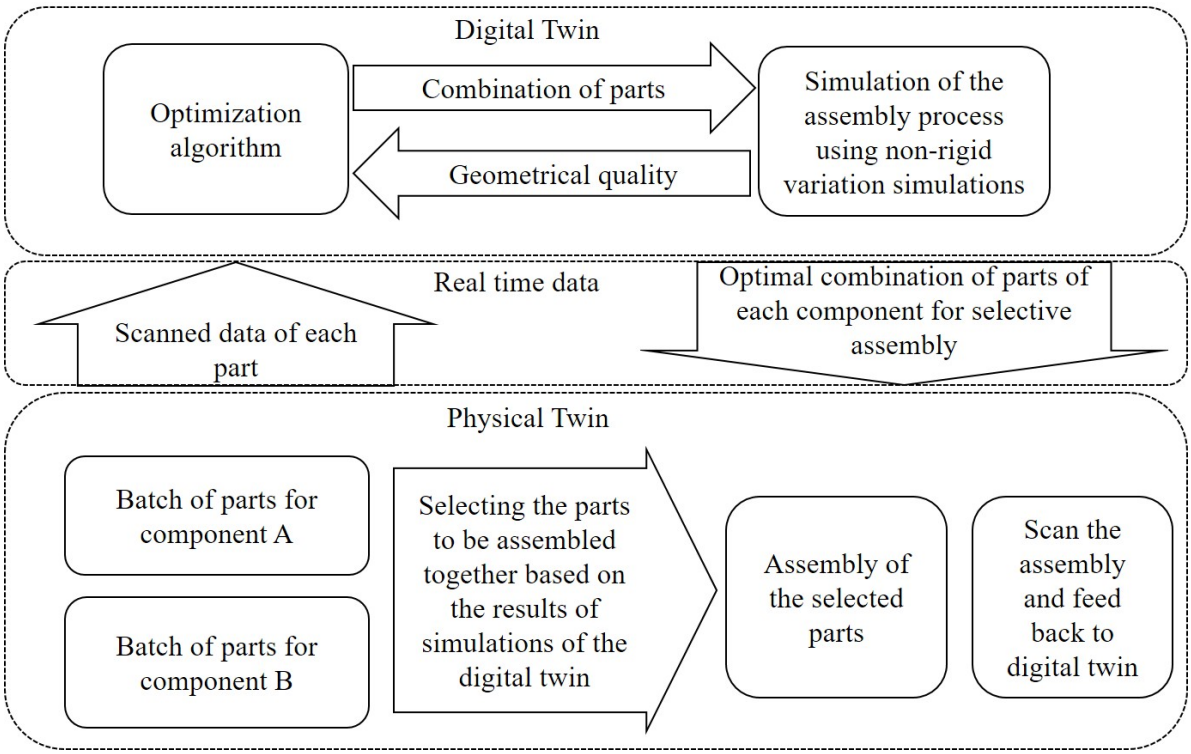

Fig. 1: An schematic diagram of a digital twin based smart assembly line of sheet metals that utilizes selective assembly technique to improve the geometrical quality of assemblies.

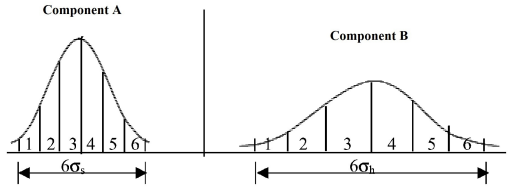

Fig. 2: Grouping of measured dimension of parts.

til the start of the assembly process. Accordingly, addressing this problem and solving it is essential for utilizing selective assembly technique in a digital twin based assembly line of sheet metals in the automotive industry.

This study presents a new method of mapping that is a one-to-one translation of phenotype to genotype and vice versa. Furthermore, the effects of using the previous and new mapping method on the performance of the optimization algorithm in solving selective assembly of three sample cases from the literature are investigated.

Section 2 reveals that the existing mapping is not one-to-one and suggests a new mapping approach to solve this problem. In Section3, three different sample cases of selective assembly problems from the literature are introduced to assess the performance of both mapping methods. The results of applying the conventional and modified mapping on the sample cases are presented and discussed in Section 4. Finally, the conclusion will be drawn from the results and discussions in the final section.

2 Method

Selective assembly is an optimization problem. The objective is usually to minimize variation of the geometrical deviations among all assemblies. For non-rigid assemblies, the mean value of the geometrical deviations can also be considered as the second objective [9]. In some studies, the cost of production is also considered as an objective [32]. Regardless of the objective, the problem is always to find the optimal combination of groups or individual parts. Therefore, selective assembly is a combinatorial optimization problem.

Metaheuristic optimization algorithms, including GA and PSO, have been previously used to solve the selective assembly problem. But there is a problem in mapping the phenotype to genotype. Considering an assembly of k components with N parts, the number of all combinations that can be used to create those assemblies is $(N!)^{(k-1)}$. For instance, consider an assembly that consists of two components, A and B ($k = 2$), where ten assemblies are going to be produced. For the first part of component A, there are ten options with which component B can be assembled. For the second component, there are nine options and so on. Therefore, all possible combinations to make those ten assemblies are 9! combinations. This paper refers to the method of mapping presented by Kannan et al. [21] as *conventional method*. The size of the produced genotype from this method of mapping will be $(N!)^{(k)}$. Hence, the conventional method maps a design domain with the size of $(N!)^{(k-1)}$ to a domain with the size of $(N!)^{(k)}$. A consequence of this type of mapping is that, during the optimizations, some answers can be produced that are different in genotype, but identical in phe-

notype. As an example consider an assembly that consists of two components and five parts ($k = 2$, $N = 5$). To generate a genotype solution using the conventional mapping method, two substrings will be created that can each take a sequence of 1 to 5, thereby producing 14400 possible solutions. However, from these solutions, 14280 solutions result in the repetitive combination of parts. For instance, the following solutions are two different answers in the genotype:

Solution 1:

5	3	4	1	2	2	5	1	4	3

Solution 2:

3	1	2	4	5	5	4	3	1	2

However, both solutions represent the following combination of parts: $(A_5B_2), (A_3B_5), (A_4B_1), (A_1B_4)$ and (A_2B_3). In this example, there are 23 additional solutions in genotypes that represent exactly this combination of parts in phenotype.

Making the genotype domain larger than the phenotype domain, as the conventional method does, may have both advantages and disadvantages. The advantage is that it may make the optimization process more exploratory [33]. As a result, the optimization algorithm has a higher chance of not being stopped in local optima and finding the global optimum. On the other hand, it may cause disruption in the optimization process [33]. Consequently, the convergence rate will decrease. Therefore, to discover whether this type of mapping would improve or worsen performance of GA, the results of the optimization that are obtained using this mapping should be compared with a one-to-one mapping method.

The new mapping presented in this paper is one-to-one. To keep the genotype size, the same as phenotype size, the number of substrings should be one less than the number of components. Therefore, the integers of the first substring represent the part number of the second component, and the second substring represents the third component and so on. The key point of this mapping is that for the first component, a fixed sequence of integers will always be considered in translation. The new mapping procedure is explained using the previous example. The combination of: (A_1B_4), (A_2B_3), (A_3B_5), (A_4B_1) and (A_5B_2) will be mapped to the following chromosome in the genotype:

4	3	5	1	2

The first integer in this chromosome represents the part number of component B that will be assembled with part number one of component A. In other words, it demonstrates that the first part from component A should be assembled with part number four of component B. The second integer shows part number three from component B should be assembled with part number two of component A and so on. This combination corresponds to only one solution in the genotype. Therefore, the size of the genotype and phenotype is exactly equal now. This mapping will be addressed as *new method* and its performance will be compared to the conventional mapping in and Section 4. Figure 3 demonstrates a sample example of phenotype and different representations of it in the conventional and new methods of mapping.

3 Sample cases

To compare the effect of the proposed method with the conventional method of mapping, both methods of mapping are applied to different selective assembly problems and the results are evaluated. There are two common types of selective assembly problems [9]; linear assemblies and sheet metal assemblies. To make the conclusions of this study robust, different sample cases from both groups of assemblies are considered. Linear assemblies are assemblies that geometrical dimensions of the product after assembly is summation or subtraction of dimensions of their mating parts. Sheet metal assemblies, one the other hand, are assemblies where the relation between geometrical quality of assembly and the mating parts before assembly is not linear. Thus, compliant variation simulations are required to predict their final dimensions. As a result, these two types of problems have different types of objective functions.

The sample case that is considered for linear assemblies is the utilized sample by Kannan et al. [21] which is also used by Ponnambalam et al. [34]. The second and third sample cases are two industrial sheet metal assemblies that are utilized by Aderiani et al. [9]. These three sample cases are presented in this section.

3.1 Linear assembly

The first sample selective assembly problem is a linear assembly of three components that is utilized by Kannan et al. [21] and Ponnambalam et al. [34] and is demonstrated in Figure 4. Consider an assembly of three components A, B and C. The final dimension of the assembly which is represented by L_4 is the sum of dimension L_1 from component A, dimension L_2 from component B and L_3 from component C. This is presented in Equation 1.

$$L_4 = L_1 + L_2 + L_3 \tag{1}$$

The produced parts of A, B and C; however, would not have the exact dimensions that are designed as L_1, L_2 and L_3, and they will vary in the range of the defined tolerances. Therefore, the real dimension of the produced assemblies can be obtained from Equation 2.

$$L_4 = L_1 + \Delta_1 + L_2 + \Delta_2 + L_3 + \Delta_3 \tag{2}$$

Where Δ_1, Δ_2 and Δ_3 are deviations of each dimension from its nominal value. This relation is shown in Figure 4.

Dimensions of the produced parts for all three components are $10.000^{+0.012}_{-0.000}$. Therefore, dimensions of the assemblies when random assembly is employed are $30.000^{+0.036}_{-0.000}$. The problem is to reduce the variation of this dimension among assemblies. It is supposed that 1000 parts are produced for each component and the produced parts of each

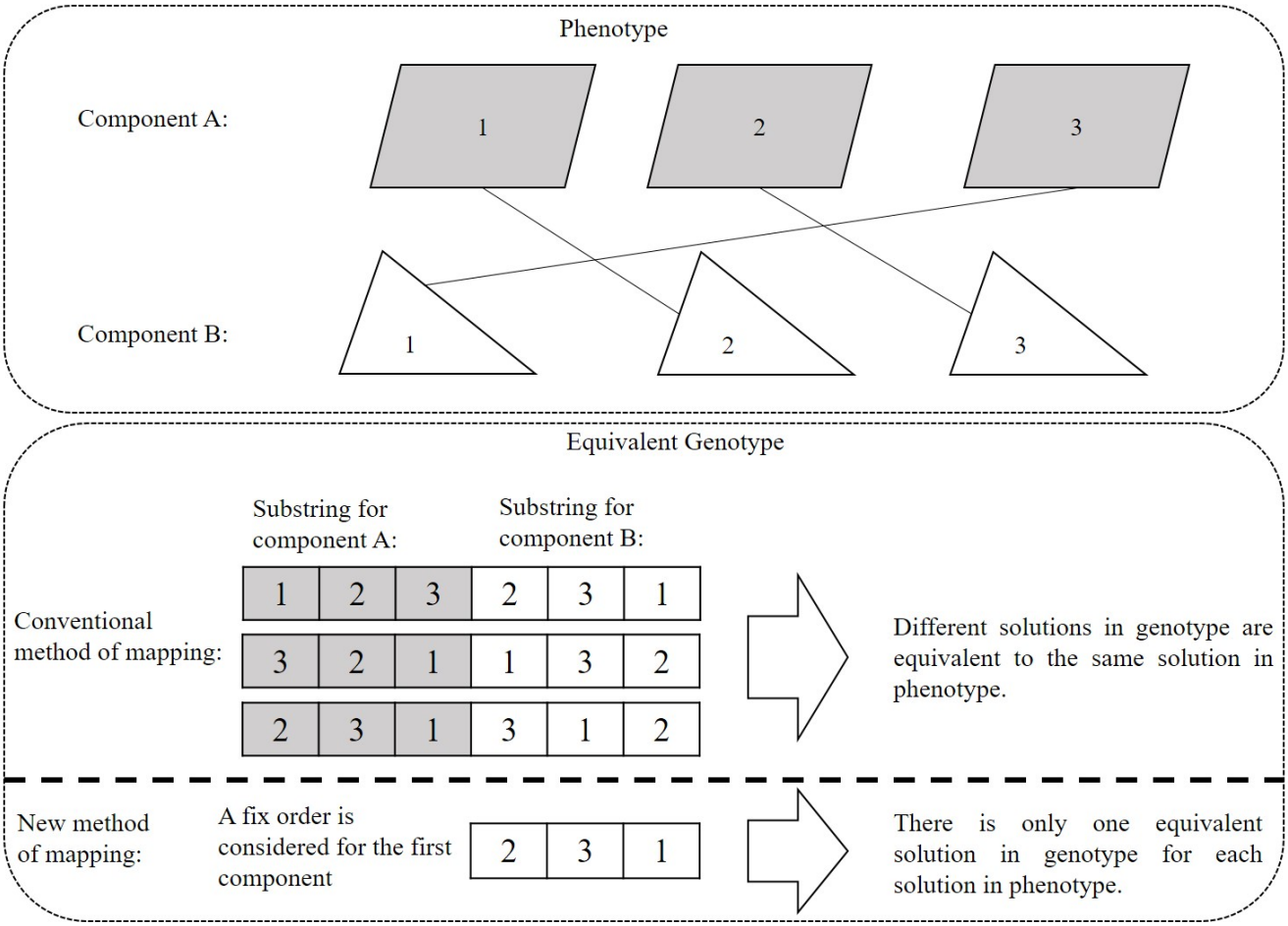

Fig. 3: A sample representation of phenotype in genotype by conventional and new methods of mapping.

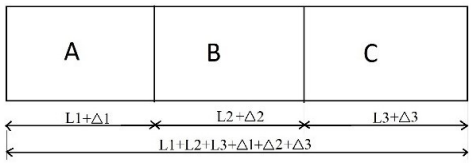

Fig. 4: First sample case, Linear assembly of three components.

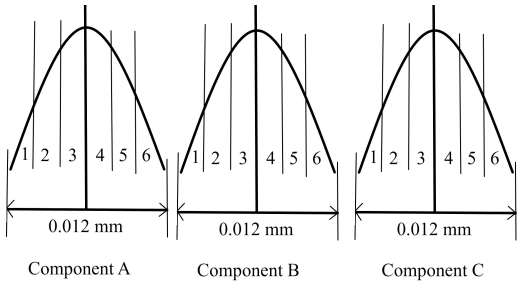

Fig. 5: Dimensional distribution of different components of the first sample case.

component are measured and divided into six groups. Since the range of part dimensions is $12\mu m$ the range of dimensions of parts in each group will be $2\mu m$. Figure 5 visualizes this grouping for each component.

The optimization problem for this sample case is to find the best combination of groups so that the difference between the maximum and minimum length of L_4 is minimal. This problem has been only solved using GA in the literature. This paper presents the mathematical formulation of this problem as a Mixed Integer Linear Programming (MILP). The results of solving this problem using branch and bound methods and GA with both new and conventional mapping will be presented in this problem. The mathematical formulation of this problem is presented by Equation 3.

$$\min w - z \tag{3}$$

Subject to:

$$w \geq (i+j+k)2x_{ijk}$$

$$z \leq (i+j+k-3)2x_{ijk}$$

$$\sum_{j}^{N}\sum_{k=1}^{N} x_{ijk} = 1; \quad i = 1,2,3\ldots 6$$

$$\sum_{i}^{N}\sum_{k=1}^{N} x_{ijk} = 1; \quad j = 1,2,3\ldots 6$$

$$\sum_{i}^{N}\sum_{j}^{N} x_{ijk} = 1; \quad k = 1,2,3\ldots 6$$

$$x_{ijk} \in \{0,1\}$$

In this formulation, w and z represent L4 of the assemblies with the maximum and minimum lengths, respectively. The group numbers of components A, B and C are indicated by i, j and k, respectively, and x_{ijk} is the optimization variable than can be zero and one. The maximum length of the parts of each group can be calculated by multiplying the group number and the tolerance range of the groups. The minimum length can also be determined for multiplying the tolerance range and the group number minus one. The group range of $2\mu m$ is multiplied to the group numbers to obtain the tolerance limits of the groups.

To solve this problem using GA, several crossover operations have been introduced for this type of optimization. Random Keys crossover [35] is chosen for application in this paper. After selecting two parents for crossover operation, these parents will be encoded by random numbers. After that, the crossover will be conducted on random numbers and two children will be generated. Finally, the children are decoded to integers. The detailed procedure is illustrated by Bean et al. [35].

A mutation is an operation with which to avoid the algorithm to converge in local minima. If a gene is selected for mutation, to perform the mutation, its location will be substituted by the following gene. Consider a chromosome such as 256143 134526 561243, if only the second gene were selected for mutation, the chromosome after mutation would be 2**65**143 134526 561243. The mutated integers are shown in bold. The cross over and mutation rates are considered to be 0.8 and 0.05, respectively [21].

3.2 Sheet metal assemblies

To compare the effect of the proposed method with the conventional method on the selective assembly of sheet metals, the second and third sample cases are selected from this type of assembly. By changing the combination of the mating parts the mean values of deviations of different points will be changed, in addition to their variation, in sheet metal assemblies [9]. Therefore, the selective assembly of sheet metals is a multi-objective optimization problem. The first objective is the Root Mean Square (RMS) of variation of deviations (RMS_v) and the second objective is the RMS of the mean value of deviations (RMS_m). Depending on the assembly, RMS_v and RMS_m can be determined for only KPC, for all nodes, or for a weighted sum of KPCs and other nodes. In the sample cases presented all nodes are considered for this aim.

The selective assembly problem for sheet metals is solved in [9] using Mixed Integer Non-linear Programming (MINLP) methods. Equation 4 presents the formulation of this problem for an assembly of two components.

$$min(\sqrt{\frac{1}{n}(\sum_{k=1}^{n}(\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N}(d_{ijk}x_{ij}))^2)}, \sqrt{\frac{1}{n}(\sum_{k=1}^{n}(\frac{36}{N-1}\sum_{i=1}^{N}\sum_{j=1}^{N}(x_{ij}(d_{ijk}-\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N}(d_{ijk}x_{ij}))^2))))} \tag{4}$$

Subject to:

$$\sum_{j}^{N} x_{ij} = 1; \quad i = 1,2,3\ldots N$$

$$\sum_{i}^{N} x_{ij} = 1; \quad j = 1,2,3\ldots N$$

$$x_{ij} \in \{0,1\}$$

In this formulation, d_{ijk} is the magnitude of the deviation in k^{th} KPC of the assembly that is generated from i^{th} part of the first component and j^{th} part of the second component. The batch size is indicated by N, and n is the number of KPCs. The optimization variable is also represented by x_{ij} in which can take zero and one. Accordingly, if x_{ij} is obtained as one, the part number i from the first component of the assembly will be assembled with the part number j of the second component. By solving the optimization problem, the matching parts from each component that result in minimal RMS_v and RMS_m of the entire batch can be obtained.

Nevertheless, since this problem is a combinatorial optimization problem, it is not practical to solve it using MINLP methods for large batch sizes. Accordingly, meta-heuristic methods including Non-dominated Sorting Genetic Algorithms (NSGA II) are superior in solving it [36]. This study utilizes both MINLP methods and an NSGA II to compare the results. However, other alternatives including PSO can also be investigated. The reason for utilizing NSGA II in this

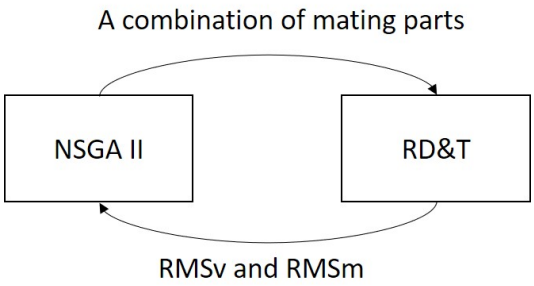

Fig. 6: Procedure of function evaluation for finding the optimal combination of parts in the second and third sample cases.

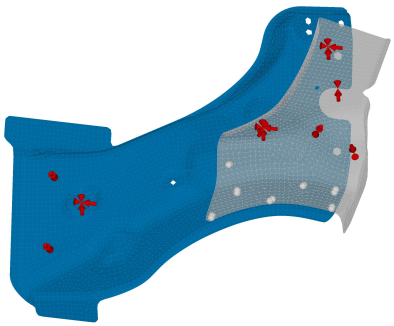

Fig. 7: The second sample case, sheet metal assembly of two components.

study is that it has been utilized relatively in more studies and there are more tools available to use it compared with other meta-heuristic algorithms. Nevertheless, the presented method of transferring phenotype to genotype in this paper is not limited to GA and can be used for other meta-heuristics including PSO.

In sheet metal assemblies, the relations between assembly dimensions with dimensions of mating parts are not linear. Thus, compliant variation simulations by Computer Aided Tolerancing (CAT) tools are required to predict their final dimensions. This study utilizes the RD&T program [1] to reach this goal. Therefore, in each function evaluation, this program calculates, the fitness of the combination for the optimization algorithm. Figure 6 visualizes this procedure.

Figures 7 and 8 present the model generated from the second and third sample cases, respectively, in the RD&T program. The arrows in these figures represent the locators of the fixture used to weld the parts together and the white spheres represent the spot welds.

This paper considers a batch of 25 assemblies from the second and third sample cases for comparison of the two mapping methods. Accordingly, 25 parts are produced for every component of each sample case. The cross over and mutation rates are considered 0.6 and 0.05, respectively. Selection of individuals for cross over is also performed using deterministic tournament selection method [37]. The

[1] www.rdnt.se

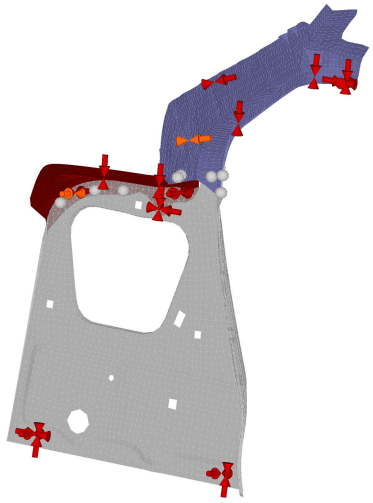

Fig. 8: The third sample case, sheet metal assembly of three components.

crossover operation is also conducted using Random Key method.

4 Results and discussions

The sample cases presented in Section 3 are solved by both the conventional and new method of mapping and the results are presented for comparison in this section. The goal of this study is to improve the convergence rate of optimization and accordingly reduce the time of calculations for the digital twin. The main bottleneck in performing the optimization, particularly in sheet metal assemblies, is the variation simulation that is required in each objective function evaluation. Therefore, the number of function evaluations that are conducted by the optimization algorithm to find a specific minimum is considered as the criterion of evaluating the convergence rate of the optimization to compare both methods of mapping.

The results are also obtained by utilizing MINLP methods and compared with the obtained results from NSGA. To utilize MINLP methods for sheet metal assemblies, for each selective assembly problem, the matrix of d_{ijk} for all i, j and k should be calculated using variation simulations before solving the problem. Accordingly, the number of function evaluations (variation simulations of assemblies) for each problem is fixed and equals to $i \times j \times k$. After determining the matrix of d_{ijk} , the optimization problem is formulated and solved using GAMS program. DICOPT, CPLEX and CONOPT solvers are employed in this tool together for obtaining the optimal solution of the problem.

The performance of GA can vary in each run due to the random factors affecting it. Nevertheless, the trends when the same settings are used in different trials can be employed to evaluate the performance. Hence, each problem is solved 100 times by each mapping method and the mean number of function evaluations is considered to make the conclusions more robust. An important factor in GA that may affect the results is population size. Thus, the performance of each

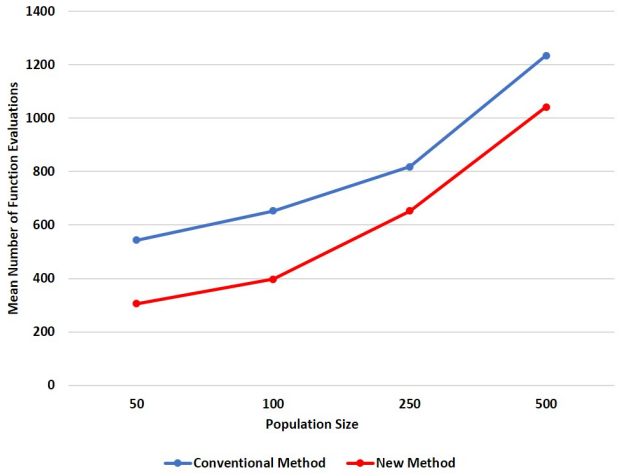

Fig. 9: Mean number of function evaluations for different population sizes in Sample case 1.

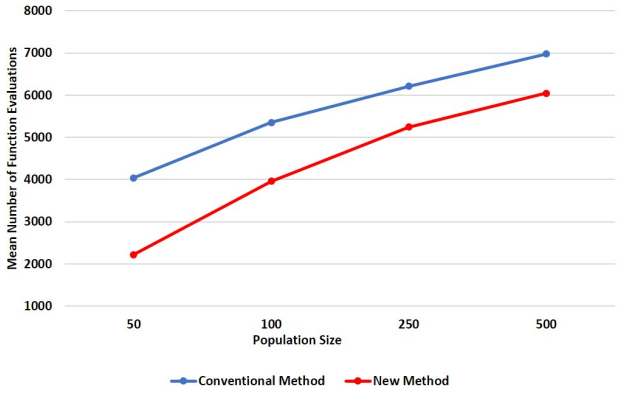

Fig. 10: Mean number of function evaluations for different population sizes in sample case 2.

coding for each sample case is assessed for four different population sizes, 50, 100, 250 and 500.

4.1 Sample case 1

The objective of optimization in this sample case is to minimize the range of L_4 in the resulting assemblies. This range is 36 μm when the mating parts are assembled randomly to each other. The minimal range that can be obtained by utilizing selective assembly is 8 μm [21]. GA is used 100 times for each population size using both methods of mapping to find the combination of groups that results in the variation of 8 μm and the number of function evaluations are recorded in each optimization. Figure 9 presents the number of objective function evaluations for different population sizes in an average for 100 replications for both conventional and the new method of mapping.

The results evidence that utilizing the new method of mapping improves the convergence speed up to almost 50% for this sample case. Moreover, the improvement is greater for smaller population sizes relative to larger population sizes.

This problem is also solved by the MILP solver of MATLAB using branch and bound method. The global optimum of 8μ is also found in this method. However, the number of nodes to be explored during branch and bound to find this optimal (function evaluations) is 20667. Comparing this number with the number of function evaluations in GA which is less than 1300, evidences superiority of meta-heuristics, particularly GA in solving these problems. The elapsed times to find the optimal solutions for the GA and the MILP method are 0.09 and 2.68 seconds, respectively.

4.2 Sample cases 2 and 3

The geometrical quality of the sheet metal assemblies when mating parts are picked randomly is obtained by simulating the assembly process for these assemblies using 10000 random combinations of parts. Thereafter, a goal is set for improvement of each criterion to be achieved by conducting selective assembly. Table 1 lists the value of each geometrical quality criterion when the random assembly is employed and the considered goals to be achieved by the selective assembly.

Table 1: RMS_v and RMS_m for random assembly and the defined goals to be achieved by selective assembly.

	Random Assembly		Goal	
	RMS_v	RMS_m	RMS_v	RMS_m
Sample case 2	0.75	0.26	0.35	0.20
Sample case 3	1.7	0.4	0.80	0.22

Figure 10 visualizes the mean number of function evaluations in 100 replications of the optimization procedure for each population size in Sample case 2. Based on the results, the mean number of function evaluations when the new method of mapping is utilized is considerably lower compared to the conventional method. The same results are presented in Figure 11 for Sample case 3. The reduction in this sample case is lower compared with Sample case 2.

The results evidence an improvement in the number of function evaluations using the new method of mapping. Figure 12 demonstrates the achieved improvements for all sample cases in different population sizes. Based on the results, the achieved improvement is larger for smaller sample sizes.

As presented by Equation 4 selective assembly of sheet metals is an MINLP problem. To evaluate the results of GA with other methods, sample cases 2 and 3 are also solved using MINLP methods. For each sample case, the matrix of d_{ijk} for all i, j and k is calculated by RD&T, firstly. Then, the optimization problem is formulated and solved using the GAMS program. DICOPT, CPLEX, and CONOPT solvers are employed in this tool together for obtaining the optimal

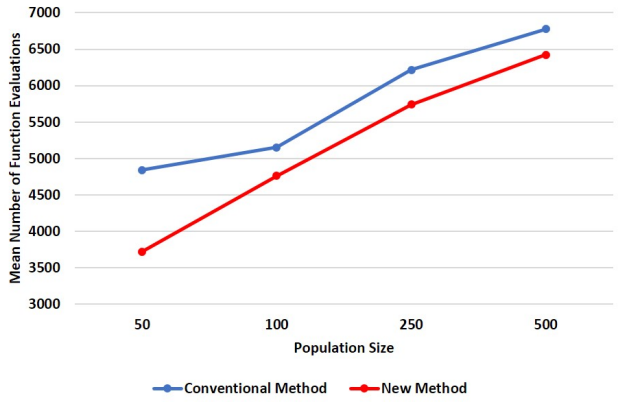

Fig. 11: Mean number of function evaluations for different population sizes in sample case 3.

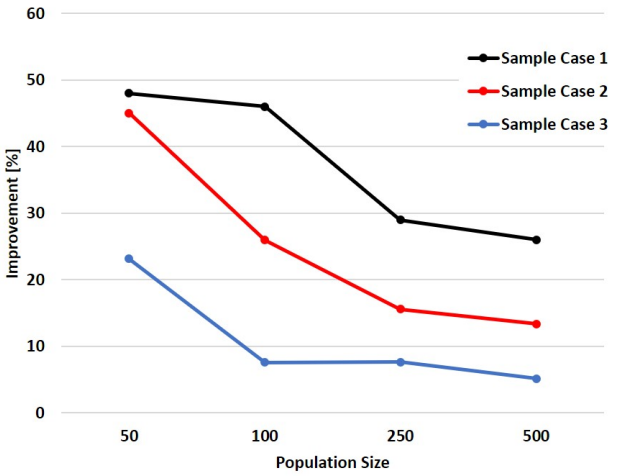

Fig. 12: Percentages of improvements attained by using the new method of mapping for different sample cases.

solution to the problem.

The size of d_{ijk} for Sample case 2 is $25^2 \times 5579 = 3486875$ and $25^2 = 625$ of variation simulation are required to determine this matrix. The number of required variation simulations to calculate d_{ijkl} for the third sample case is $25^3 = 15625$ and its size is $25^3 \times 4708 = 73562500$. To obtain the two objectives for this problems $\varepsilon - constraint$ method is utilized. Accordingly, RMS_v is considered as a constraint and RMS_m is determined. The obtained RMS_m for the presented goals of RMS_v in Table 1 are 0.195 and 0.22, respectively.

Each function evaluation in Sample case 2 and 3 takes 30 and 46 seconds, respectively, using a Core i7 CPU and 16 MB of RAM in a PC. Accordingly, utilizing the new method of mapping can save up to 16 and 26 hours in the optimization procedure for the second and third sample cases, respectively. The approximate elapsed time to solve the second sample case using MINLP methods, including variation simulations, is 6 hours. This time is approximately 15 hours using GA with the new coding. These times for Sample case 3 are 190 and 44 for MINLP and GA with the new coding, respectively. Accordingly, by increasing the number of components or the batch size, the number of variables in the MINLP method increases exponentially and meta-heuristic methods including GA are superior in solving the problem. Utilizing MINLP methods for assemblies of more than two components and a batch size of 50 is almost impractical.

4.3 Future research

Applying selective assembly on designs with slip planes is a potential subject for future studies. The focus of this study has been on GA. Therefore, the difference in the performance of other evolutionary algorithms based on the same concept of mapping can be assessed in future studies. In addition, the performance of these two different mapping method for assemblies with more than three components can also be studied in future researches. Furthermore, utilizing other techniques to improve the speed of optimization including utilizing surrogate modeling can be studied in future studies.

5 Conclusion

This study addresses a principle problem in using selective assembly technique in a digital twin based assembly line of sheet metal assemblies in the automotive industry. This problem is the calculation cost of the optimization process and consequently the time of finding the optimal combination of parts. It is revealed that the existing method of mapping from phenotype to genotype in solving the selective assembly problem using an evolutionary algorithm is not an injective or one-to-one function. A new method of mapping is presented to make the mapping one-to-one. Thereafter, the performance of both types of mappings for three sample cases is evaluated. The mean number of objective function evaluations in 100 replications to find a specific goal is considered as the criterion to evaluate the performance of each method of mapping. Moreover, the problem is solved using MINLP methods for evaluation of the results.

The results indicate that meta-heuristic methods including GA are superior in solving the selective assembly problem compared with MILP and MINLP methods particularly for assemblies with more than two components and batch sizes of more than roughly 50.

The results evidence that using the new method of mapping improves the convergence rate significantly for all sample cases. This improvement is greater in smaller populations sizes where it is 49%, 48% and 25% in the first, second and third sample cases, respectively. This means reducing 16 and 26 hours of the calculation time for the second and third sample cases, respectively. Moreover, between Sample case 2 and 3, sheet metal assemblies, Sample case 2 has relatively greater improvements. Therefore, it can be concluded that the achieved improvements by utilizing the new method are greater for sheet metal assemblies with a lower number of components compared with sheet metal assemblies with a larger number of components. This is because the number of different solutions in the genotype that are correspondent to one solution in the phenotype, when the conventional method is utilized, is greater for assemblies with a lower number of

components compared to assemblies with a larger number of components.

Acknowledgements

This work was carried out in collaboration within Wingquist Laboratory and the Area of Advance Production at Chalmers within the project Smart Assembly 4.0, financed by The Swedish Foundation for Strategic Research. Their support is gratefully acknowledged.

References

[1] Mansor, E. M., 1961. "Selective assembly-its analysis and applications". *International Journal of Production Research,* **1**(1), pp. 13–24.

[2] Söderberg, R., Wärmefjord, K., Carlson, J. S., and Lindkvist, L., 2017. "Toward a Digital Twin for real-time geometry assurance in individualized production". *CIRP Annals - Manufacturing Technology,* **66**(1), pp. 137–140.

[3] Colledani, M., Ebrahimi, D., and Tolio, T., 2014. "Integrated quality and production logistics modelling for the design of selective and adaptive assembly systems". *CIRP Annals - Manufacturing Technology,* **63**(1), pp. 453–456.

[4] Tao, F., and Zhang, M., 2017. "Digital twin shop-floor: A new shop-floor paradigm towards smart manufacturing". *IEEE Access,* **5**, pp. 20418–20427.

[5] Rezaei Aderiani, A., Wärmefjord, K., Söderberg, R., and Lindkvist, L., 2019. "Individualizing Locator Adjustments of Assembly Fixtures Using a Digital Twin". *Journal of Computing and Information Science in Engineering,* **19**(4), pp. 041019–041028.

[6] Cronrath, C., Aderiani, A. R., and Lennartson, B., 2019. "Enhancing digital twins through reinforcement learning". In 2019 IEEE 15th International Conference on Automation Science and Engineering (CASE), pp. 293–298.

[7] Lyu, N., Lee, B., and Saitou, K., 2005. "Optimal Subassembly Partitioning of Space Frame Structures for In-Process Dimensional Adjustability and Stiffness". *Journal of Mechanical Design,* **128**(3), 08, pp. 527–535.

[8] Lee, B., and Saitou, K., 2007. "Assembly synthesis with subassembly partitioning for optimal in-process dimensional adjustability". *Artificial Intelligence for Engineering Design, Analysis and Manufacturing,* **21**(1), p. 31–43.

[9] Rezaei Aderiani, A., Wärmefjord, K., and Söderberg, R. "Developing a Selective Assembly Technique for Sheet Metal Assemblies". *International Journal of Production Research,* **57**(22), pp. 7174–7188.

[10] Lorin, S., Lindau, B., Lindkvist, L., and Söderberg, R., 2018. "Efficient Compliant Variation Simulation of Spot-Welded Assemblies". *Journal of Computing and Information Science in Engineering,* **19**(1), 11. 011007.

[11] Lindau, B., Wärmefjord, K., Lindkvist, L., and Söderberg, R., 2014. "Method for Handling Model Growth in Nonrigid Variation Simulation of Sheet Metal Assemblies". *Journal of Computing and Information Science in Engineering,* **14**(3), 04. 031004.

[12] Desmond, D. J., and Setty, C. A., 1961. "Simplification of selective assembly". *International Journal of Production Research,* **1**(3), pp. 3–18.

[13] Fang, D., and Zhang, Y., 1995. "A new algorithm for minimising the surplus parts in selective assembly". *Computers and Industrial Engineering,* **28**(2), pp. 341–350.

[14] Fang, X. D., and Zhang, Y., 1996. "Assuring the matchable degree in selective assembly via a predictive model based on set theory and probability method". *Journal of Manufacturing Science Engineering,* **118**, pp. 252–258.

[15] Pugh, G. A., 1992. "Selective assembly with components of dissimilar variance". *Computer and Industrial Engineering,* **23**(1–4), pp. 487—491.

[16] Chan, K. C., and Linn, R. J., 1998. "a Grouping Method for Selective Assembly of Parts of Dissimilar Distributions". *Quality Engineering,* **11**(2), pp. 221–234.

[17] Coullard, C., Gamble, A., and Jones, P., 1998. "Matching problems in selective assembly operations". *Annals of Operations Research,* **76**, p. 95–107.

[18] Iwata, S., Matsui, T., and McCormick, S. T., 1998. "A fast bipartite network flow algorithm for selective assembly". *Operations research letters,* **22**(4-5), pp. 137–143.

[19] Burkard, R., Dell'Amico, M., and Martello, S., 2009. *Assignment Problems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA.

[20] Tan, M. H., and Wu, C. F. J., 2012. "Generalized selective assembly". *IIE Transactions,* **44**(1), pp. 27–42.

[21] Kannan, S., Jayabalan, V., and Jeevanantham, K., 2003. "Genetic algorithm for minimizing assembly variation in selective assembly". *International Journal of Production Research,* **41**(14), pp. 3301–3313.

[22] Kannan, S. M., Sivasubramanian, R., and Jayabalan, V., 2009. "Particle swarm optimization for minimizing assembly variation in selective assembly". *International Journal of Advanced Manufacturing Technology,* **42**(7-8), pp. 793–803.

[23] Chen, J., Dai, K. S., and Xu, S. L., 2012. "Selective assembly for components with multiple characteristics". In Automatic Manufacturing Systems II, Vol. 542 of *Advanced Materials Research*, Trans Tech Publications, pp. 79–86.

[24] Kannan, S., Sivasubramanian, R., and Jayabalan, V., 2009. "A new method in selective assembly for components with skewed distributions". *International Journal of Productivity and Quality Management,* **4**(5/6), p. 569.

[25] Rezaei Aderiani, A., Wärmefjord, K., and Söderberg, R., 2018. "A Multistage Approach to the Selective Assembly of Components without Dimensional Distribution Assumptions". *Journal of Manufacturing Science and Engineering,* **140**(7), p. 071015.

[26] Kannan, S. M., Asha, A., and Jayabalan, V., 2005. "A new method in selective assembly to minimize clearance variation for a radial assembly using genetic algorithm". *Quality Engineering,* **17**(4), pp. 595–607.

[27] Kumar, M. S., Kannan, S., and Jayabalan, V., 2007. "A new algorithm for minimizing surplus parts in selective assembly by using genetic algorithm". *International Journal of Production Research,* **45**(20), pp. 4793–4822.

[28] Kumar, M. S., and Kannan, S., 2007. "Optimum manufacturing tolerance to selective assembly technique for different assembly specifications by using genetic algorithm". *International Journal of Advanced Manufacturing Technology,* **32**(5-6), pp. 591–598.

[29] Asha, A., Kannan, S. M., and Jayabalan, V., 2008. "Optimization of clearance variation in selective assembly for components with multiple characteristics". *International Journal of Advanced Manufacturing Technology,* **38**(9-10), pp. 1026–1044.

[30] Babu, J. R., and Asha, A., 2014. "Tolerance modelling in selective assembly for minimizing linear assembly tolerance variation and assembly cost by using Taguchi and AIS algorithm". *International Journal of Advanced Manufacturing Technology,* **75**(5-8), pp. 869–881.

[31] Lanza, G., Haefner, B., and Kraemer, A., 2015. "Optimization of selective assembly and adaptive manufacturing by means of cyber-physical system based matching". *CIRP Annals - Manufacturing Technology,* **64**(1), pp. 399–402.

[32] Tan, M. H., and Wu, C. F. J., 2012. "Generalized selective assembly". *IIE Transactions,* **44**(1), pp. 27–42.

[33] Srinivas, M., and Patnaik, L. M., 1994. "Genetic algorithms: A survey". *computer,* **27**(6), pp. 17–26.

[34] Ponnambalam, S. G., Saravana Sankar, S., Sriram, S., and Gurumarimuthu, M., 2007. "Parallel populations genetic algorithm for minimizing assembly variation in selective assembly". *2006 IEEE International Conference on Automation Science and Engineering, CASE*, pp. 496–500.

[35] Bean, J. C., 1994. "Genetic Algorithms and Random Keys for Sequencing and Optimization". *ORSA Journal on Computing,* **6**(2), pp. 154–160.

[36] Blum, C., and Roli, A., 2003. "Metaheuristics in combinatorial optimization: Overview and conceptual comparison". *ACM Comput. Surv.,* **35**(3), Sept., pp. 268–308.

[37] Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T., 2002. "A fast and elitist multiobjective genetic algorithm: NSGA-II". *IEEE Transactions on Evolutionary Computation,* **6**(2), apr, pp. 182–197.